동영상 스트리밍 서버 구현기

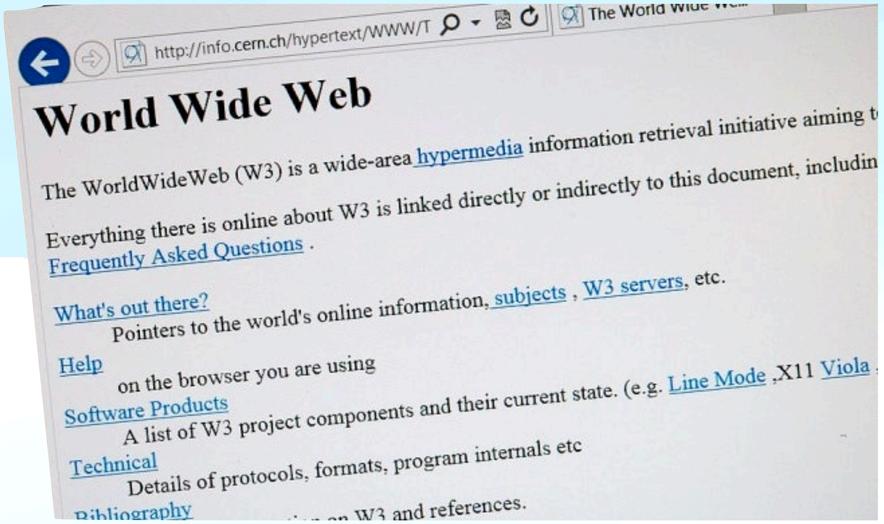
개발 과정 및 향후 개선 방향

20180237 김온지

스트리밍 기초 개념

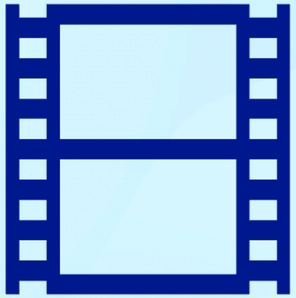
스트리밍의 등장

초기의 웹사이트

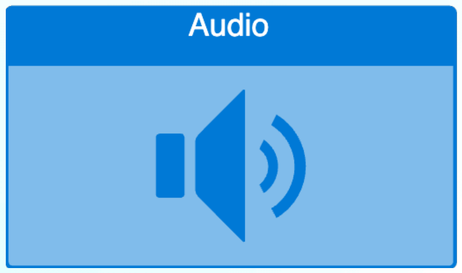


스트리밍의 등장

오늘날 웹 리소스

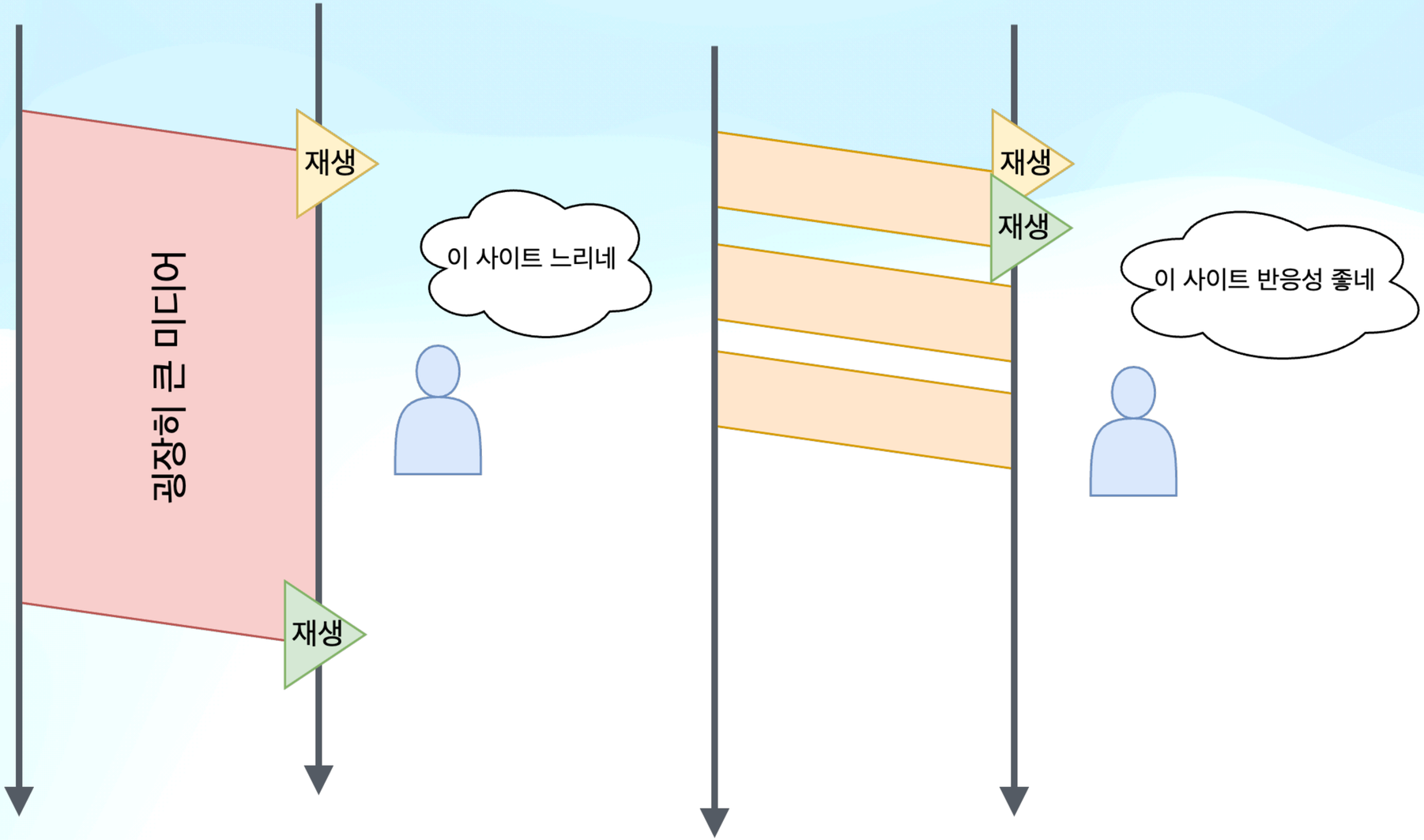


- html 파일과 비교가 안되는 큰 용량
- 실시간으로 변하는 데이터
- 다운로드를 통한 리소스 이용의 불편함



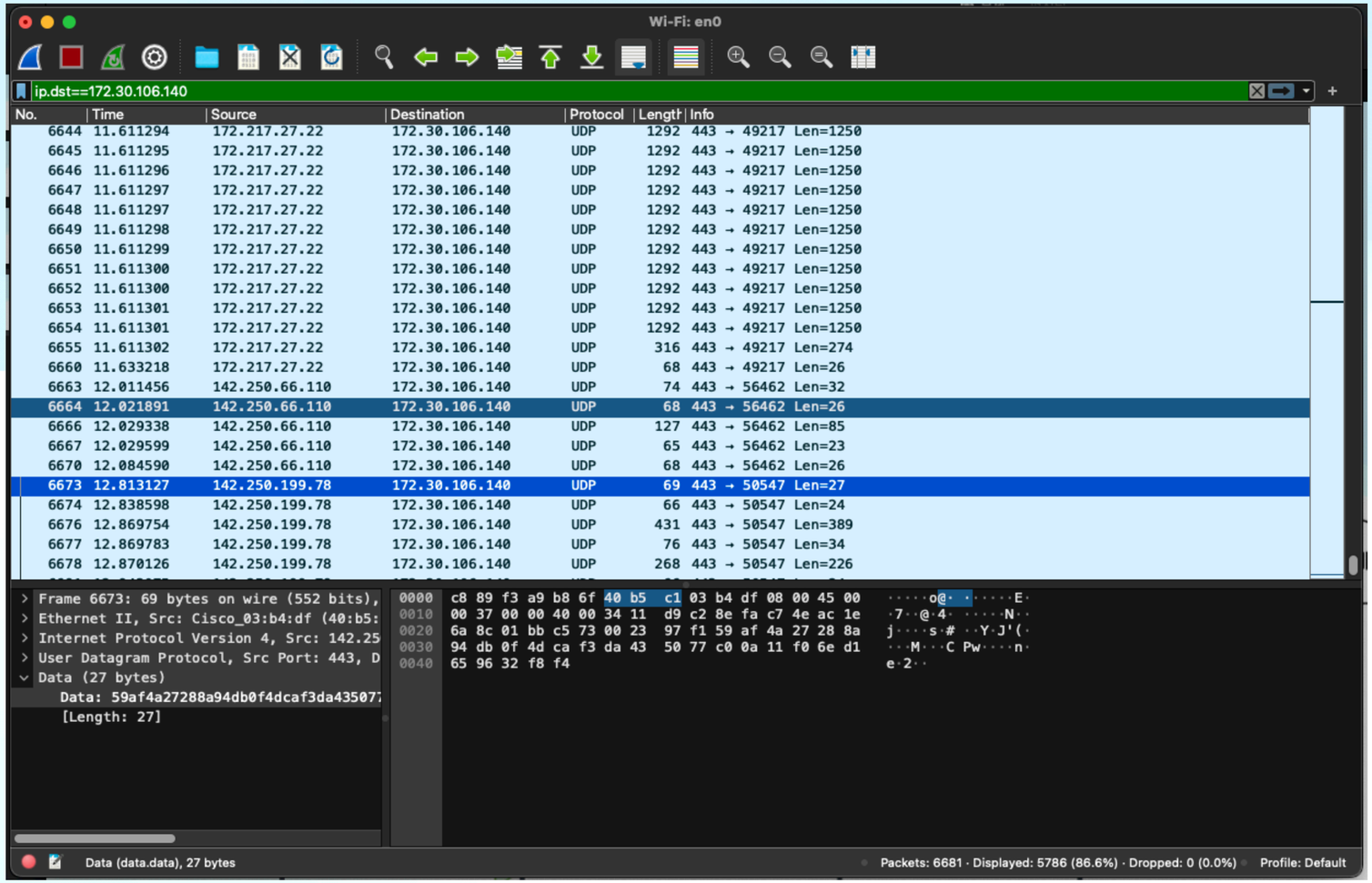
스트리밍의 등장

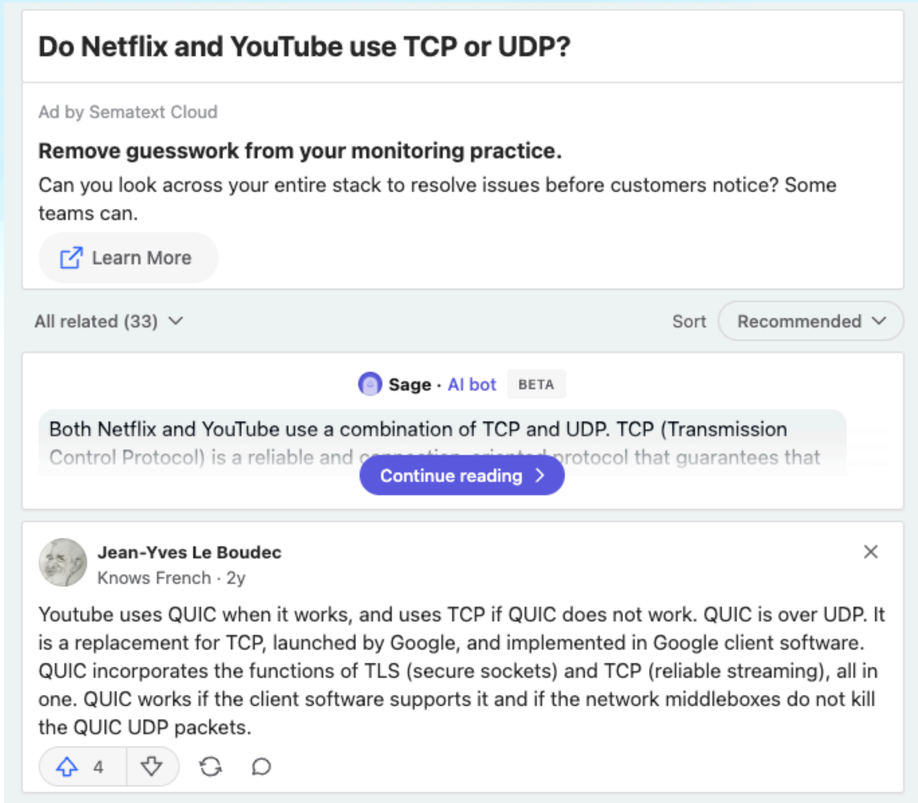
다운로드와 스트리밍의 차이점



TCP? UDP?

상황에 맞게 둘다 사용





비디오 스트리밍 구현

첫번째 시도

Java Spring FrameWork의 Resource Region 사용

Package org.springframework.core.io.support

Class ResourceRegion

java.lang.Object
    org.springframework.core.io.support.ResourceRegion

```
public class ResourceRegion
extends Object
```

Region of a `Resource` implementation, materialized by a `position` within the `Resource` and a byte `count` for the length of that region.

Since:

4.3

Author:

Arjen Poutsma

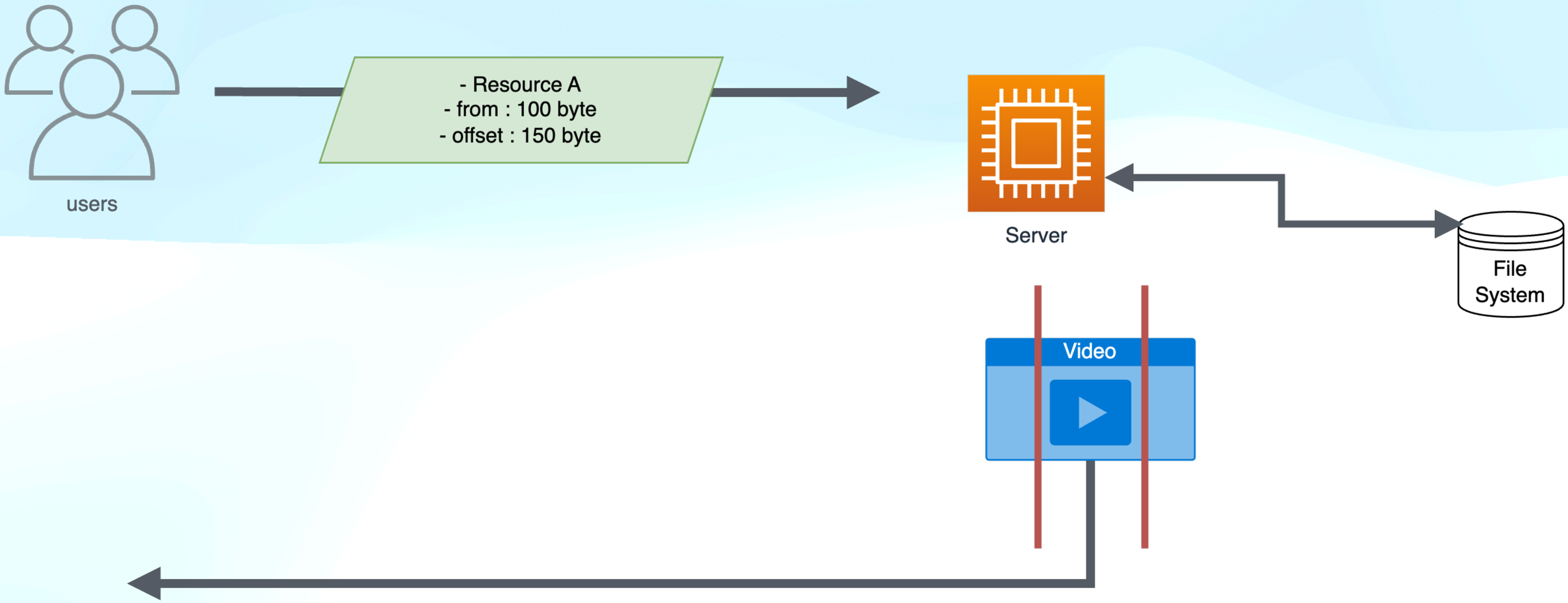
Constructor Summary

Constructors

Constructor	Description
`ResourceRegion(Resource resource, long position, long count)`	Create a new ResourceRegion from a given Resource.

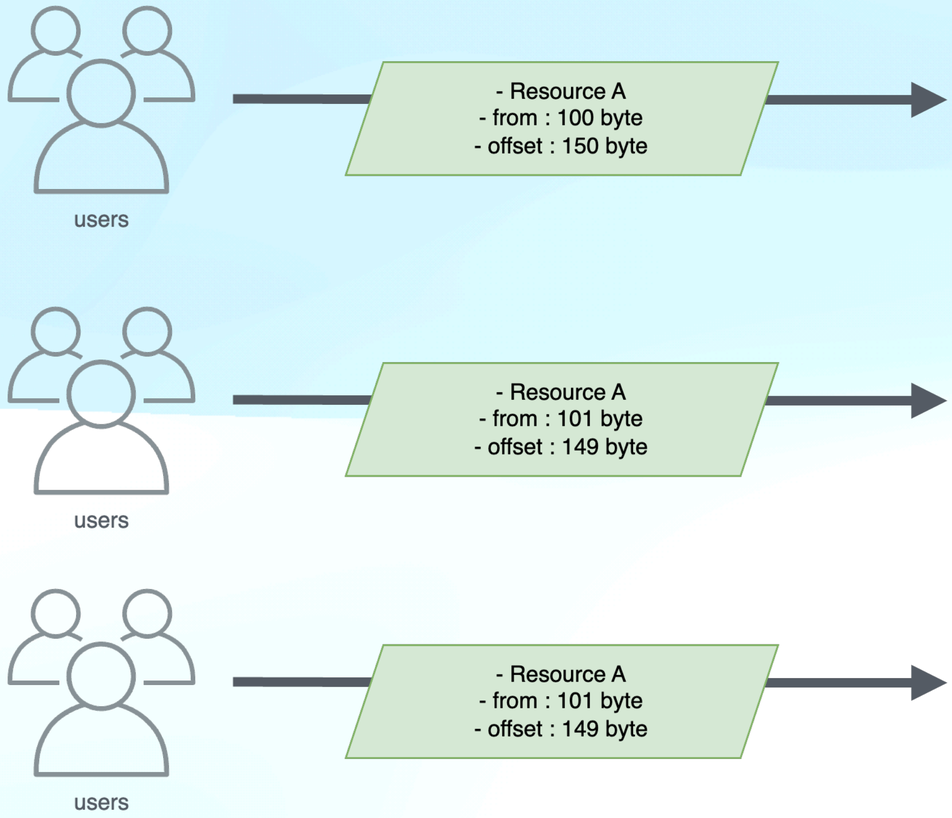
첫번째 시도

Java Spring FrameWork의 Resource Region 사용



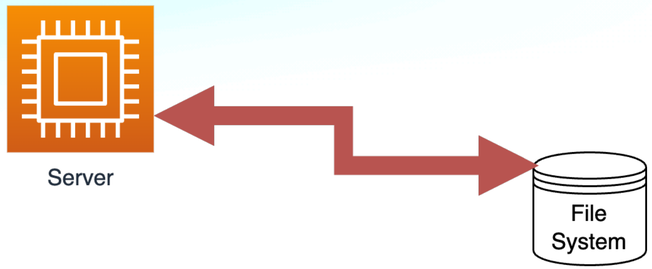
첫번째 시도

Java Spring FrameWork의 Resource Region 사용



캐싱이 어렵다

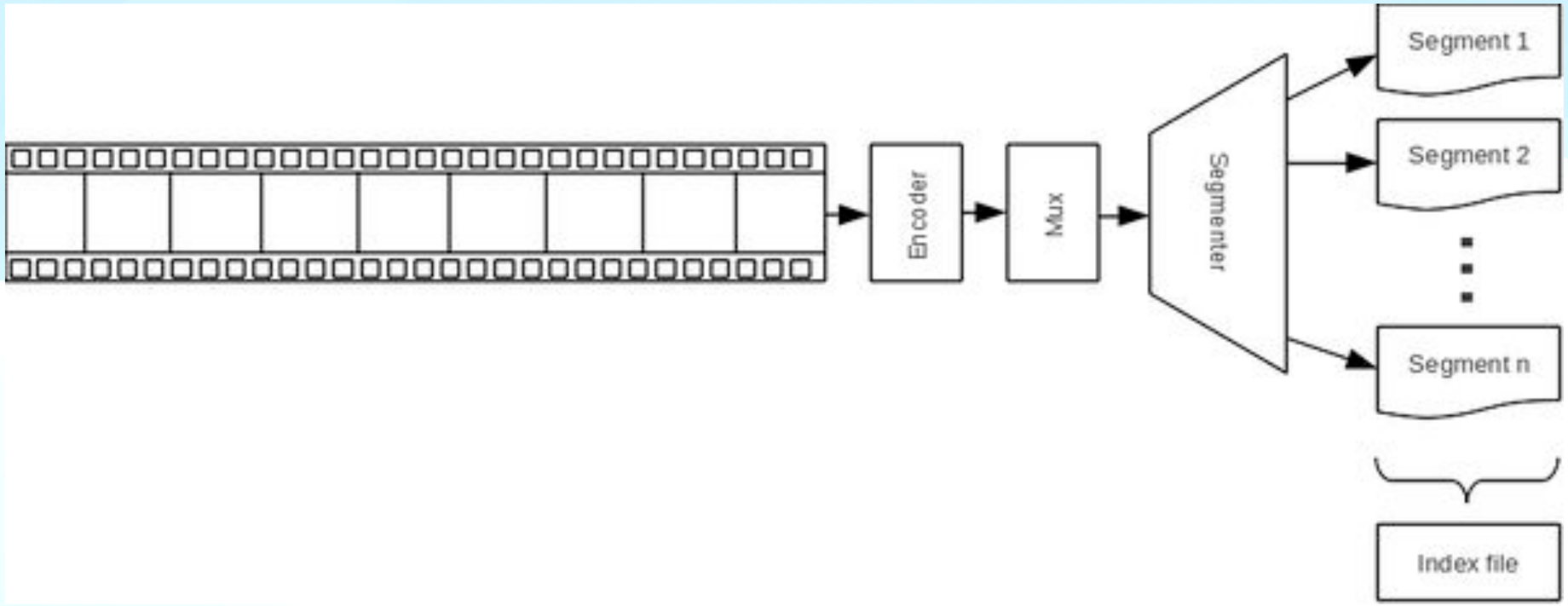
파일 시스템을 너무 빈번하게 접근해야한다.



동일한 영상 조각에 대해서도, 매번 수행해야한다.

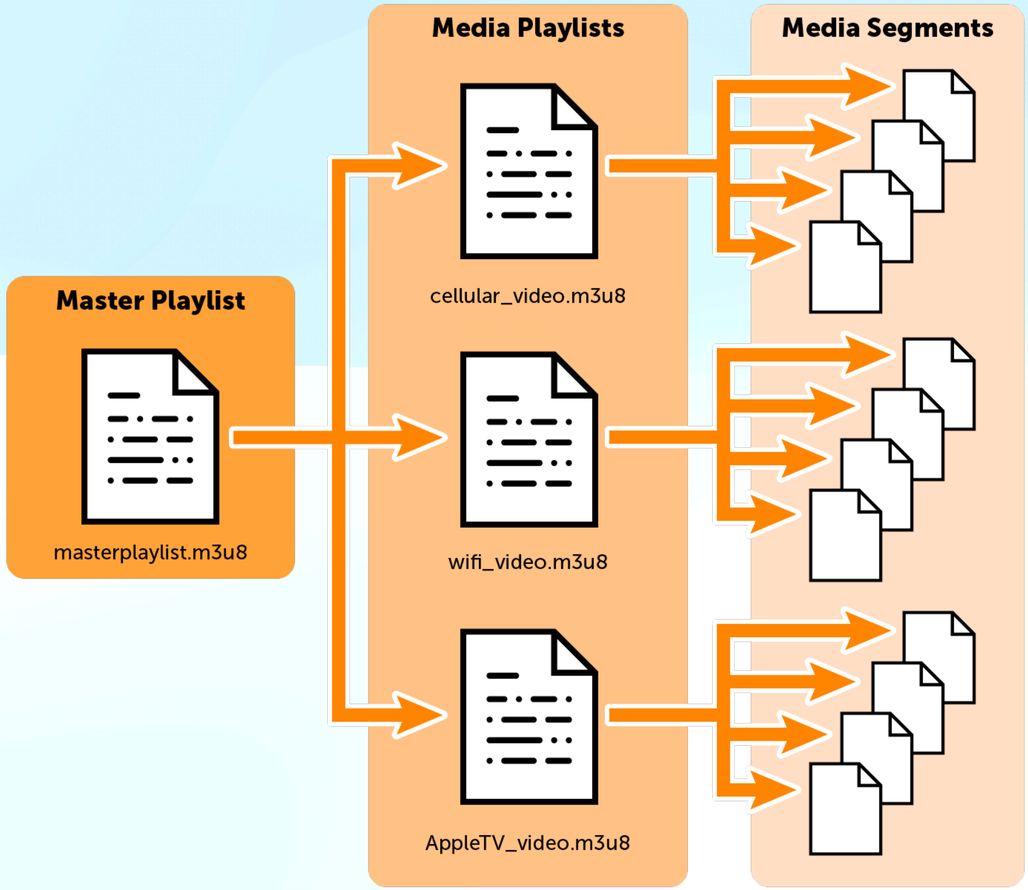
스트리밍 성능 개선

HLS(Http Live Streaming) 프로토콜로 전환



스트리밍 성능 개선

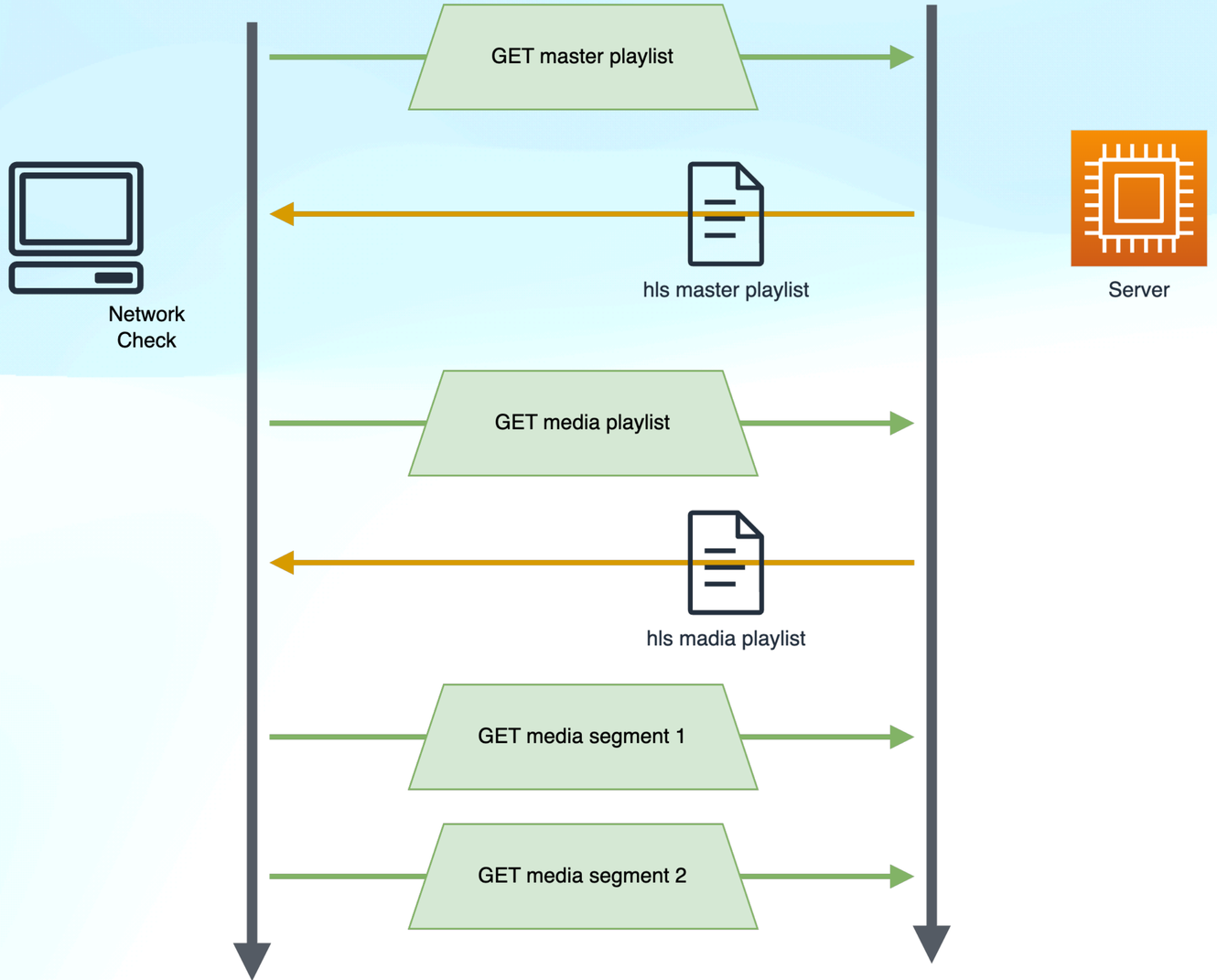
HLS(Http Live Streaming) 프로토콜로 전환




```
#EXTM3U
#EXT-X-PLAYLIST-TYPE:VOD
#EXT-X-TARGETDURATION:10
#EXT-X-VERSION:4
#EXT-X-MEDIA-SEQUENCE:0
#EXTINF:10.0,
http://example.com/movie1/fileSequenceA.ts
#EXTINF:10.0,
http://example.com/movie1/fileSequenceB.ts
#EXTINF:10.0,
http://example.com/movie1/fileSequenceC.ts
#EXTINF:9.0,
http://example.com/movie1/fileSequenceD.ts
#EXT-X-ENDLIST
```

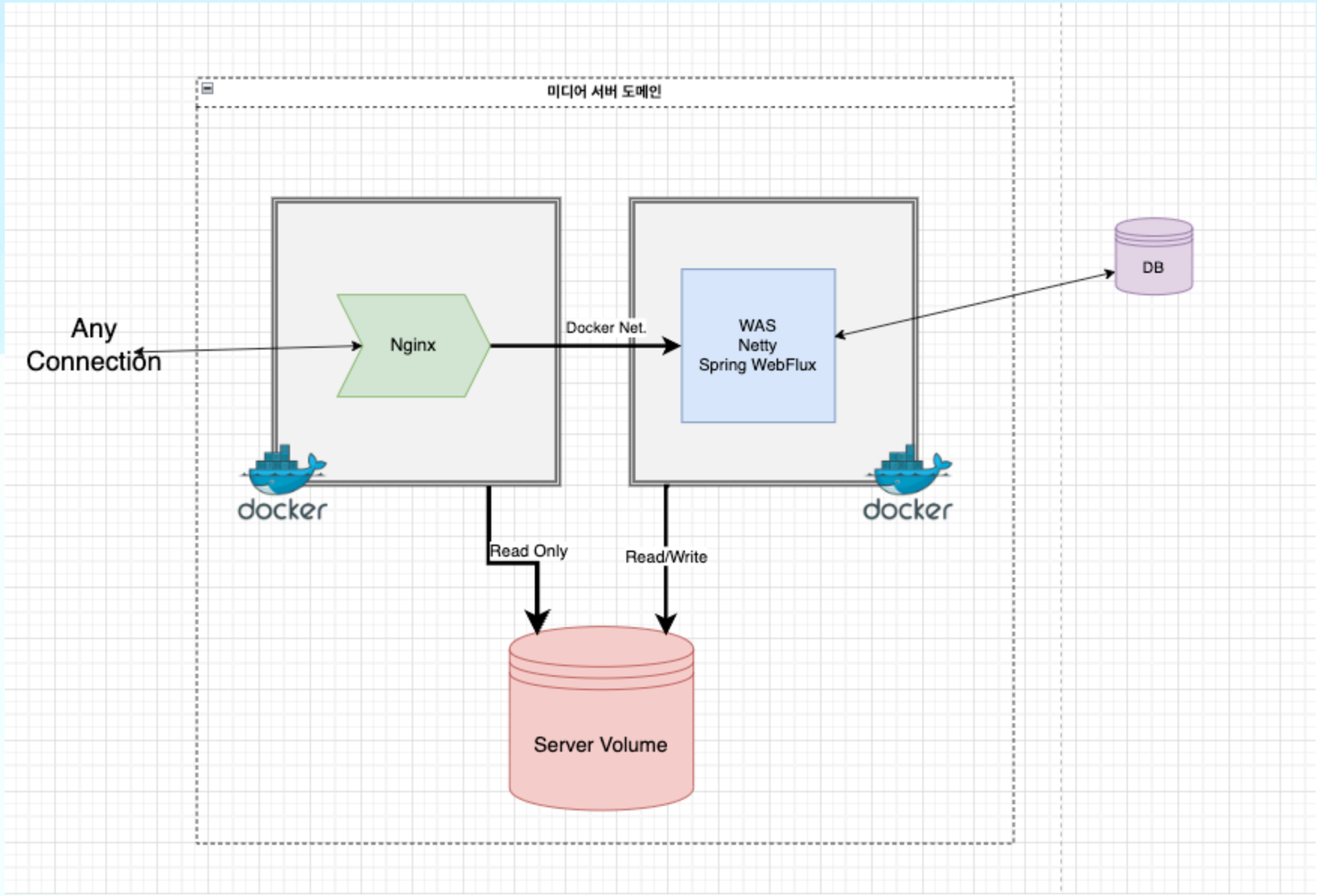
스트리밍 성능 개선

HLS(Http Live Streaming) 프로토콜로 전환



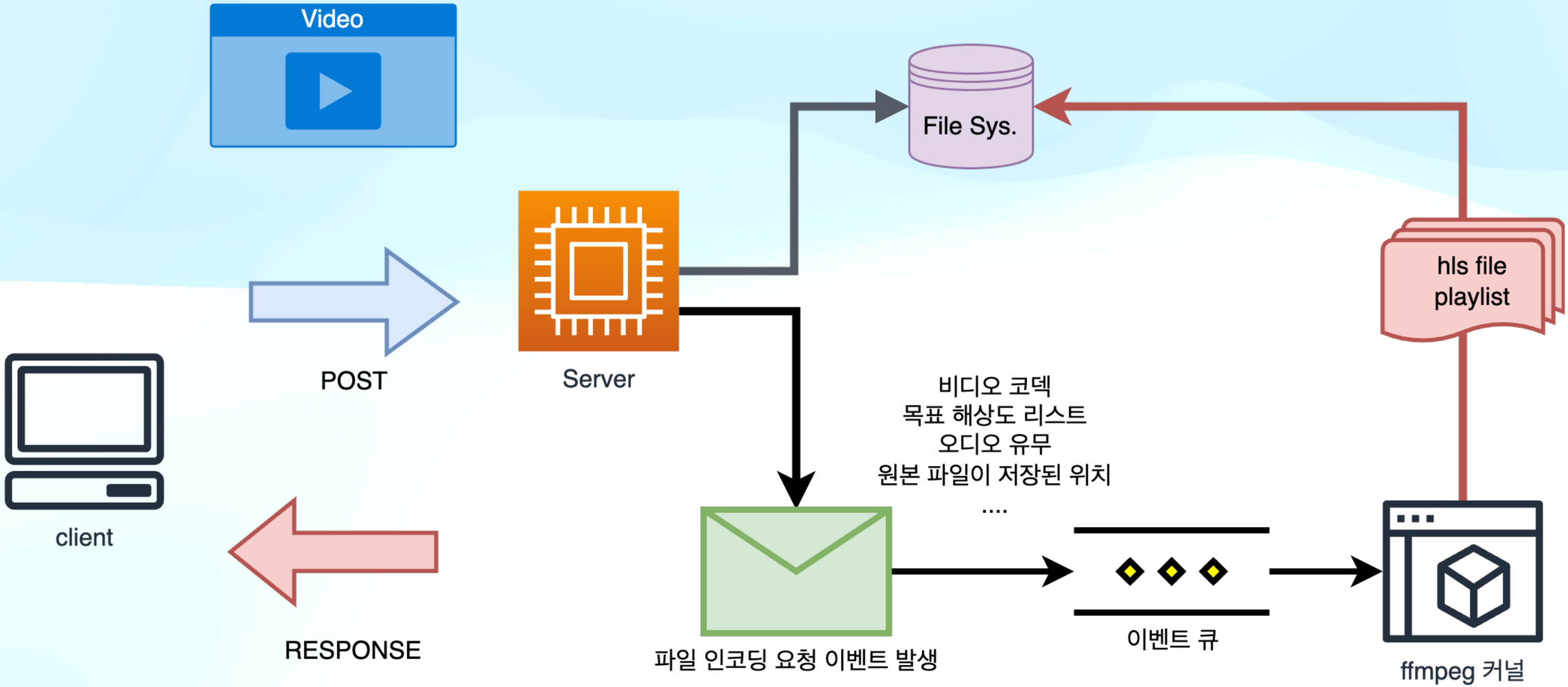
스트리밍 성능 개선

HLS(Http Live Streaming) 프로토콜로 전환



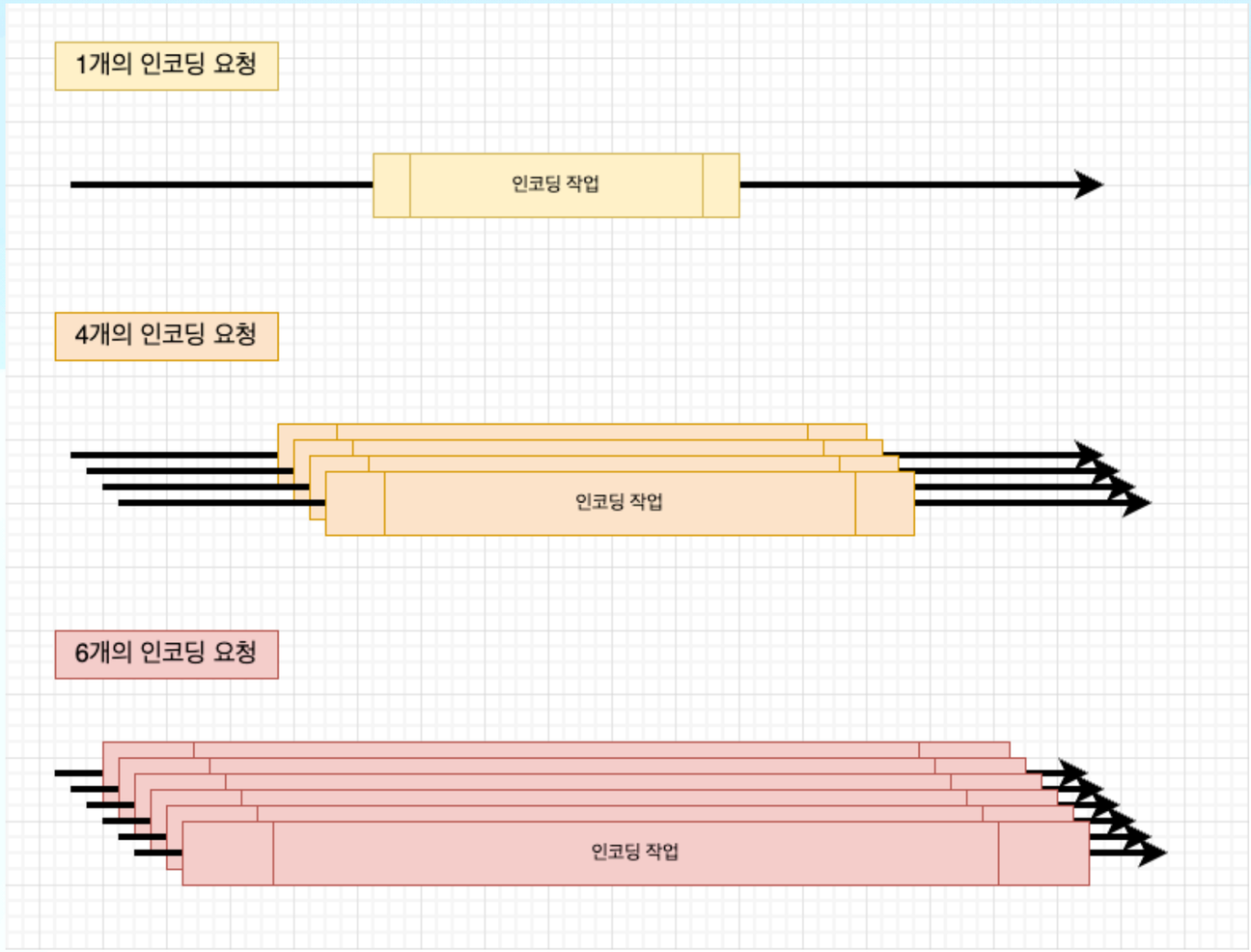
업로드 로직

비동기로 인코딩을 하도록 처리



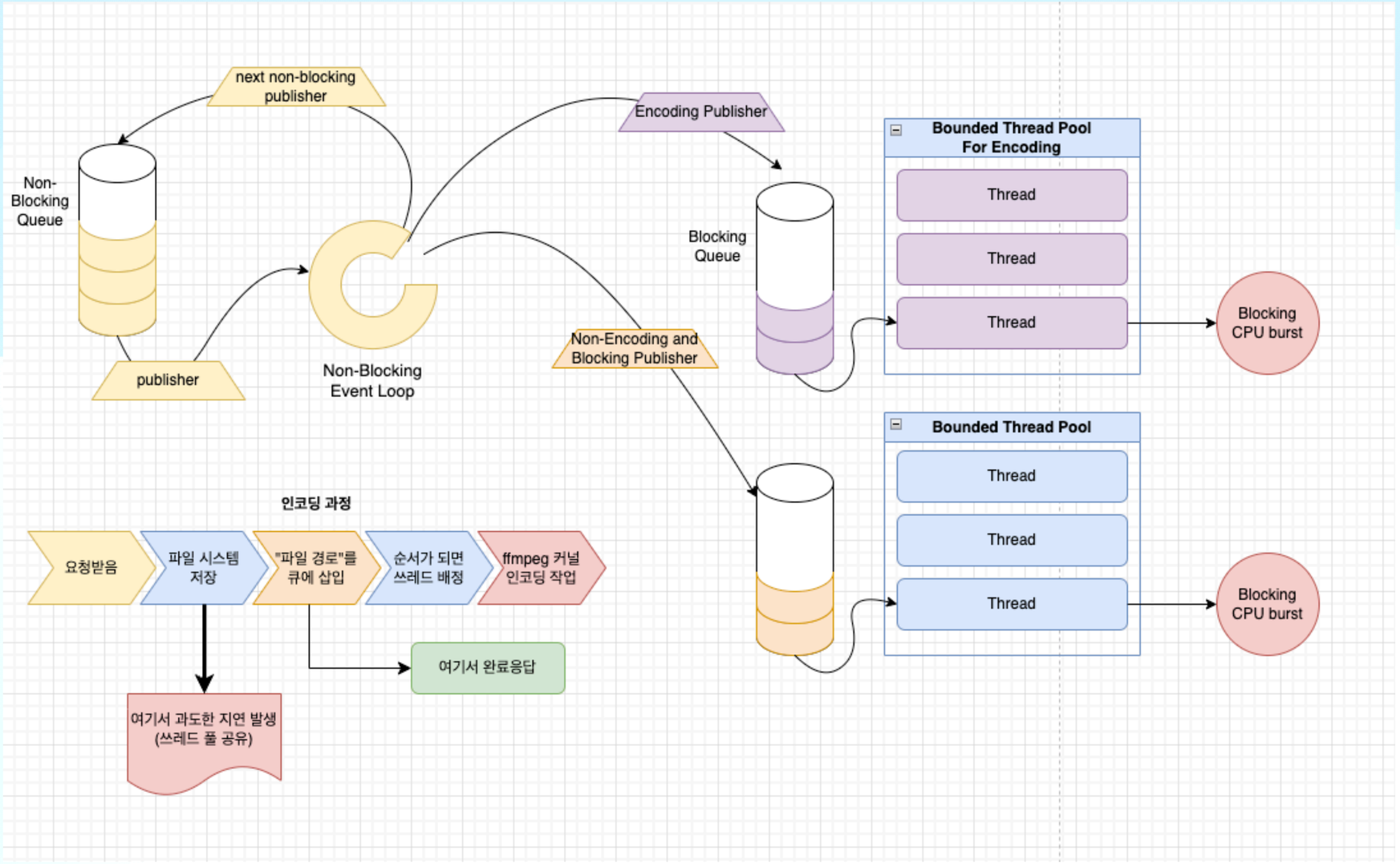
업로드 로직

인코딩 커널 프로세스 / 요청 담당 쓰레드를 무작정 늘릴 수 없다



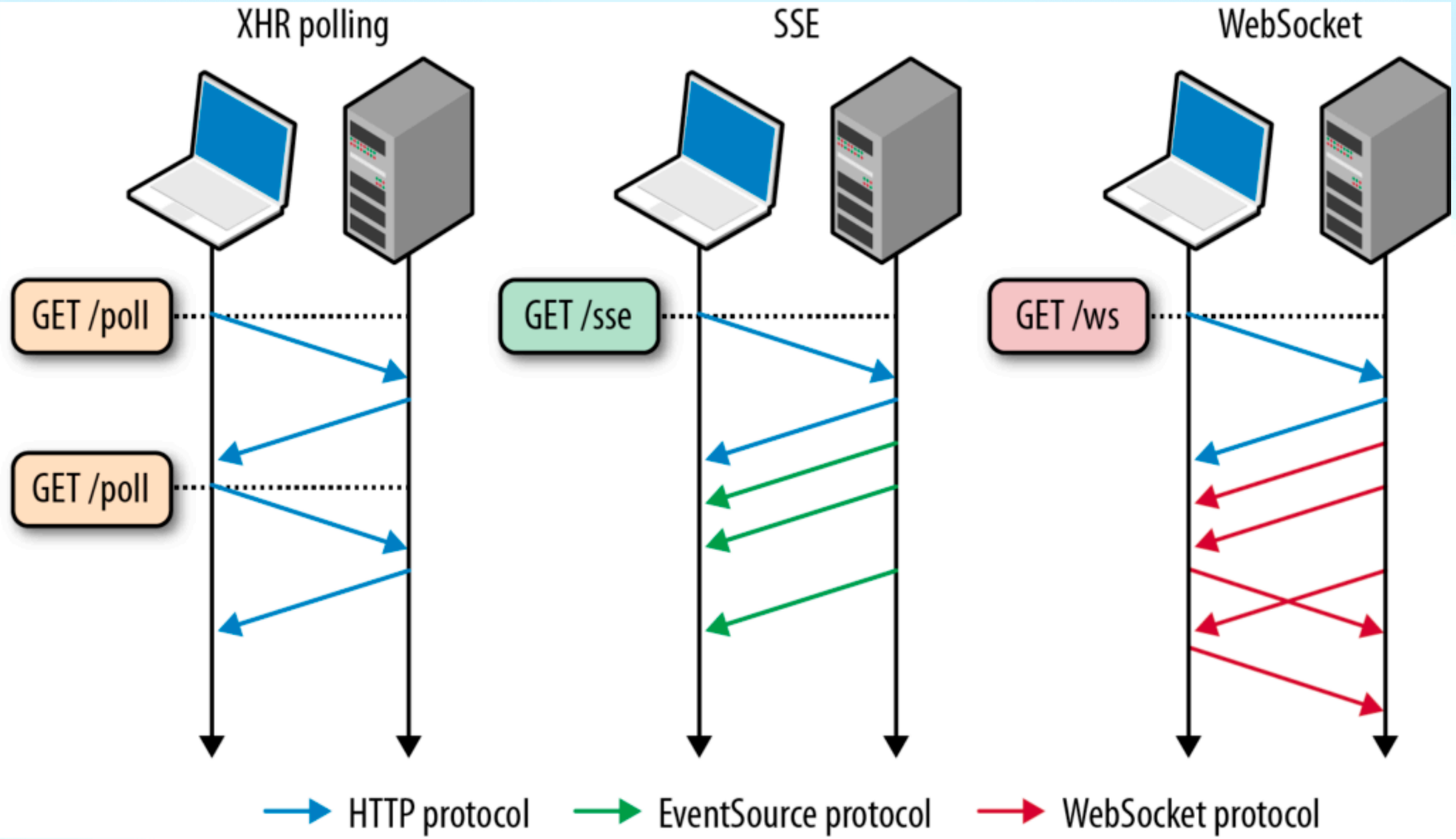
업로드 로직

인코딩 커널 프로세스를 무작정 늘릴 수 없다



비동기 인코딩의 진행상황을 어떻게 알 수 있을까?

네트워크 부분



비동기 인코딩의 진행상황을 어떻게 알 수 있을까?

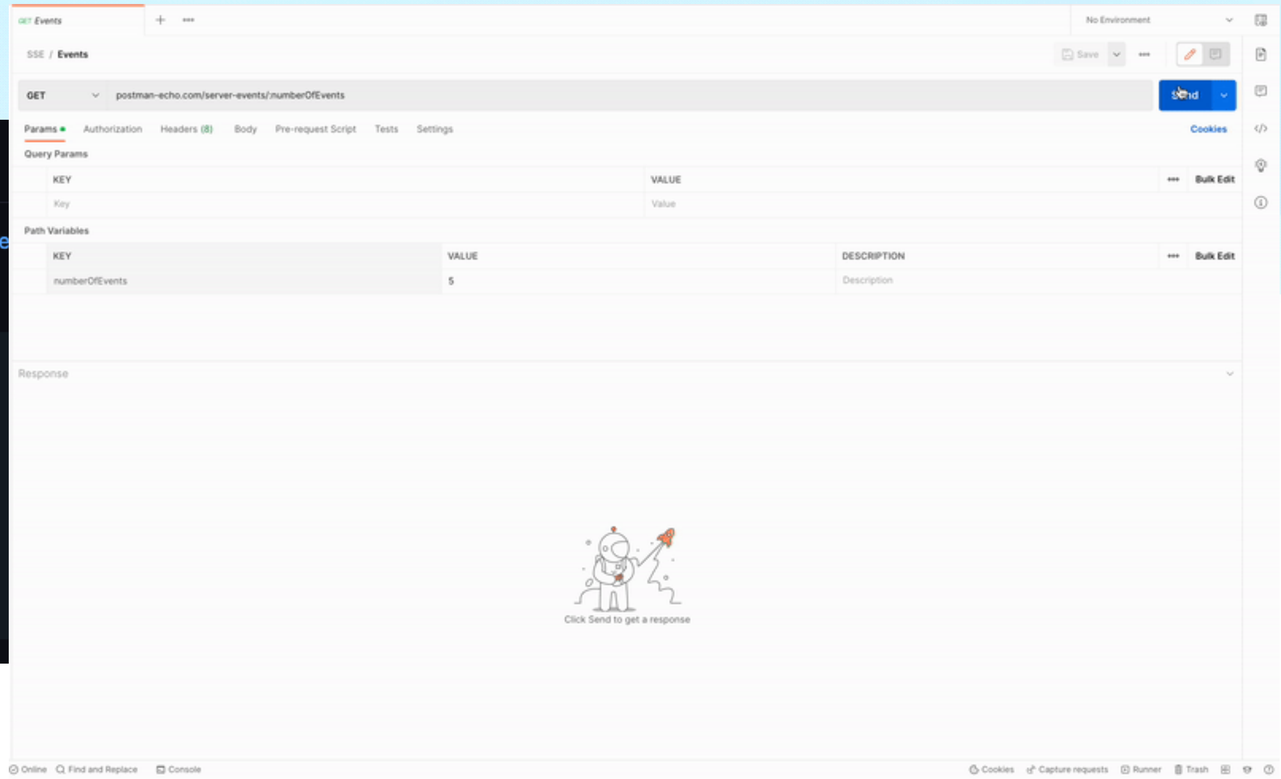
네트워크 부분

GET /media/vods/{videoId}/encoding/statuses

비디오 진행 상황을 SSE로 중계 받는 api (https://developer.mozilla.org/ko/docs/Web/API/Server-sent_eve

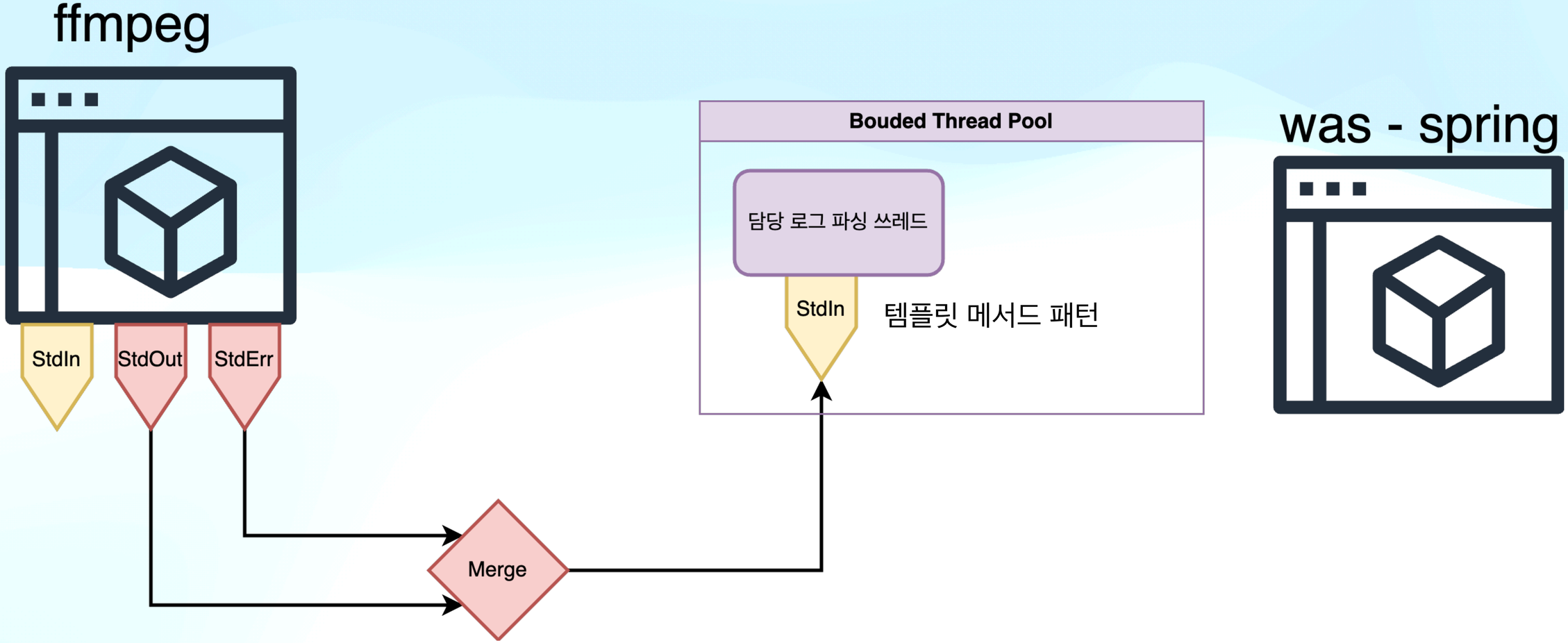
한줄씩 소수점 2자리 float으로 전송됨 (아래는 예시)

```
2.33
12.87
23.75
35.07
46.50
57.71
69.81
81.47
93.34
100.00
```



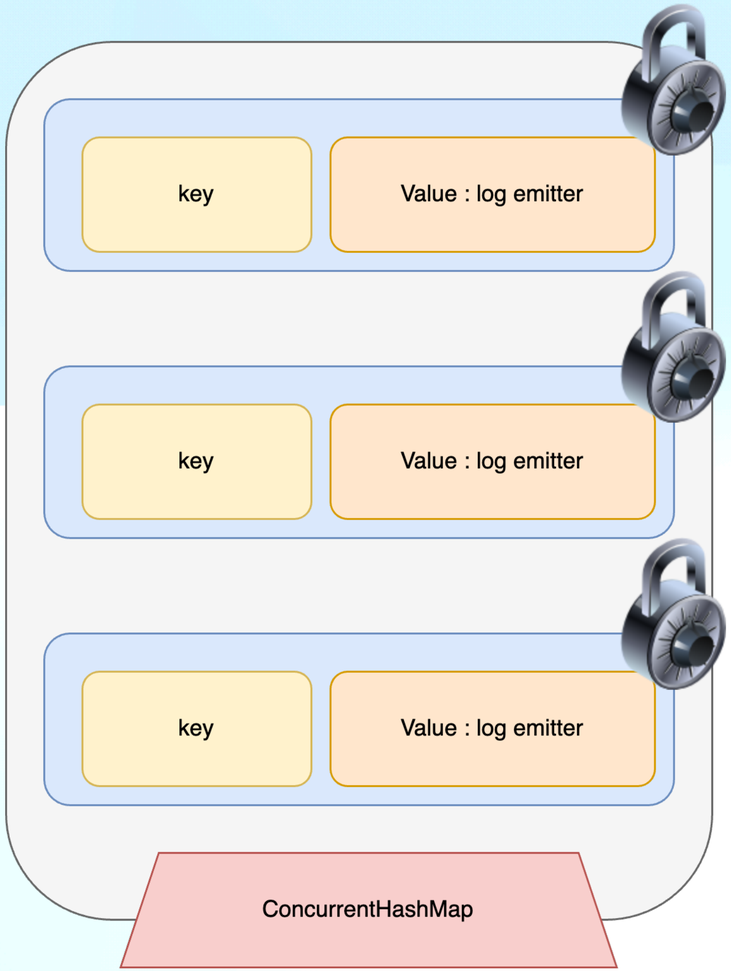
비동기 인코딩의 진행상황을 어떻게 알 수 있을까?

인코딩 프로세스 - 서버 프로세스간 연결



비동기 인코딩의 진행상황을 어떻게 알 수 있을까?

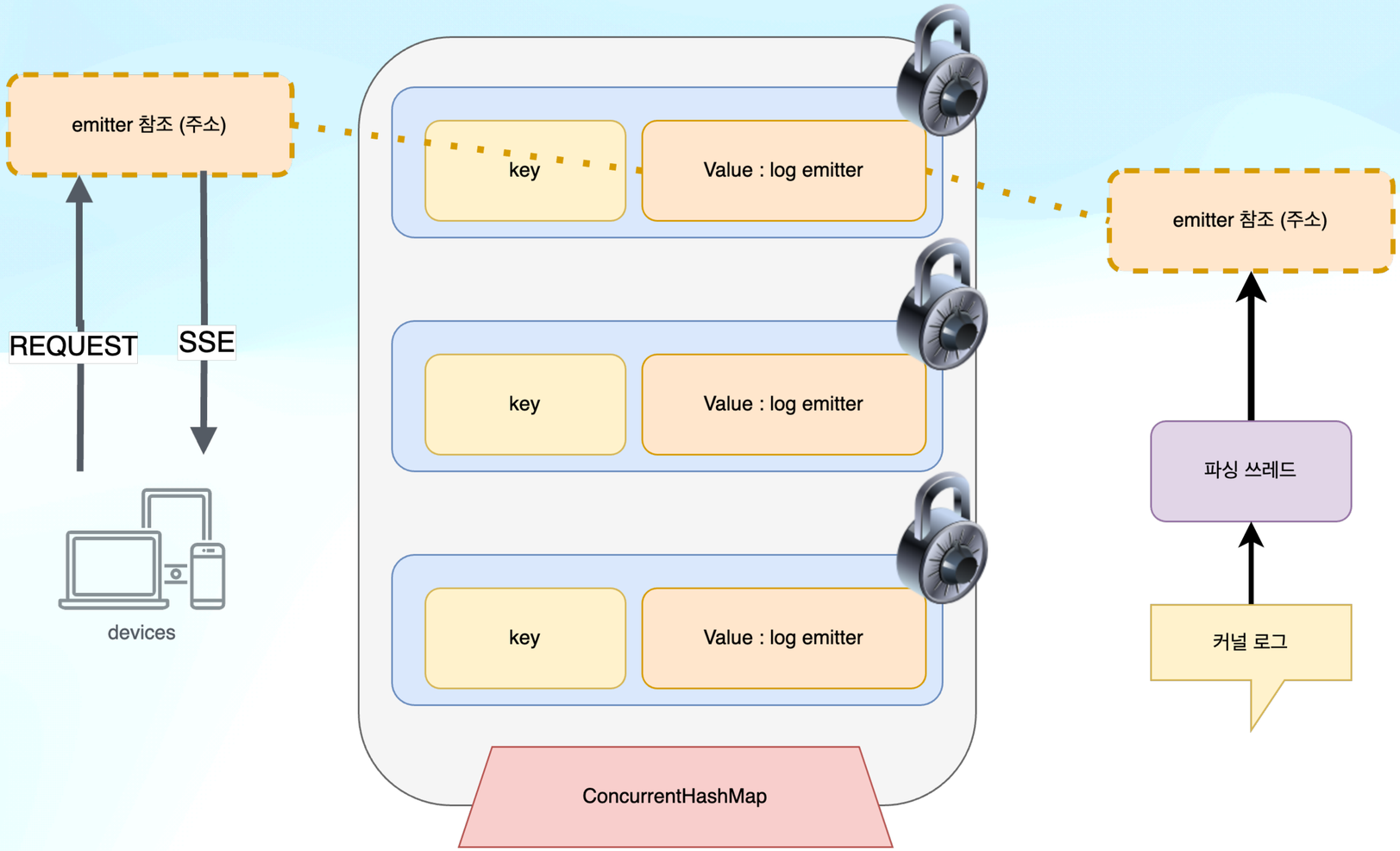
진행 상황 중계 API




```java
public class EncodingEvent<T> {
    public enum Status {
        READY,RUNNING,COMPLETE,ERROR
    };
    private final Sinks.Many<T> sink;
    private volatile Status status;

    //예외시 실행할 폴백
    private final Runnable failureHandler;
    private final Runnable completeHandler;


    public EncodingEvent(Sinks.Many<T> sink, Runnable failureHandler, Runnable completeHandler) {
        this.sink = sink;
        this.status = Status.READY;
        this.failureHandler = failureHandler;
        this.completeHandler = completeHandler;
    }

    public EncodingEvent(Sinks.Many<T> sink) {
        this.sink = sink;
        this.status = Status.READY;
        this.failureHandler = () -> {};
        this.completeHandler = () -> {};
    }

    public Flux<T> getFlux(){
        return sink.asFlux();
    }
    public Status getStatus() {
        return status;
    }

    public void reportRunning(){
```

비동기 인코딩의 진행상황을 어떻게 알 수 있을까?

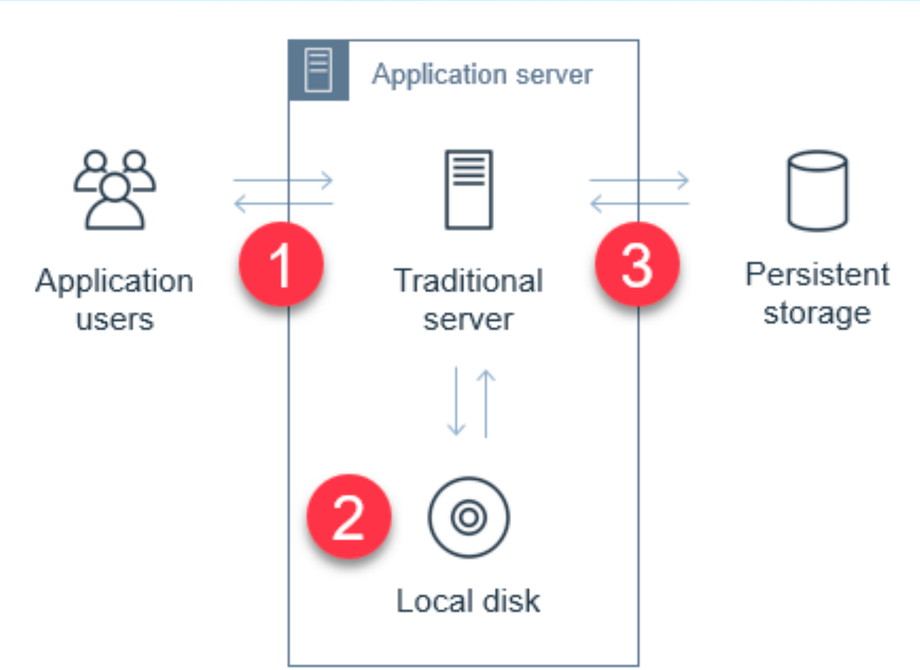
진행 상황 중계 API

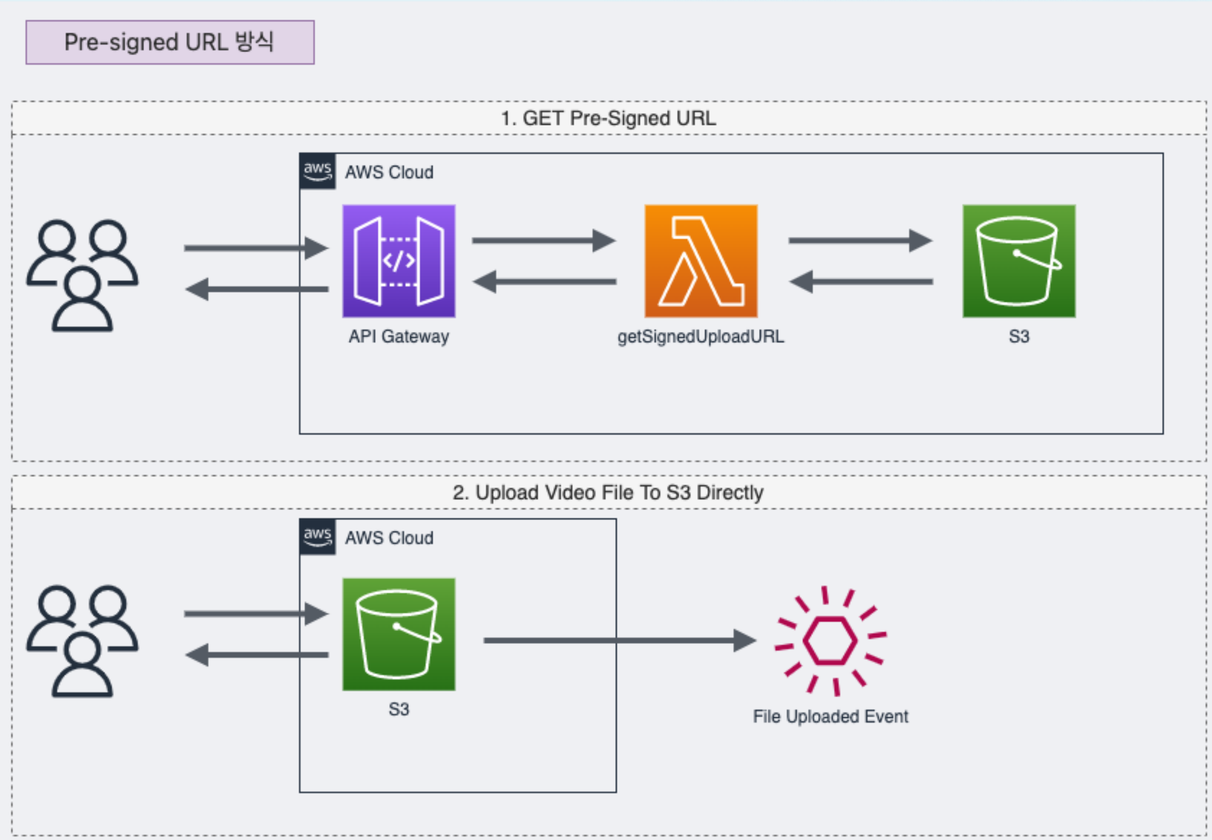


추후 개선/공부 방향

추후 개선 예정 사항

동영상 업로드 개선

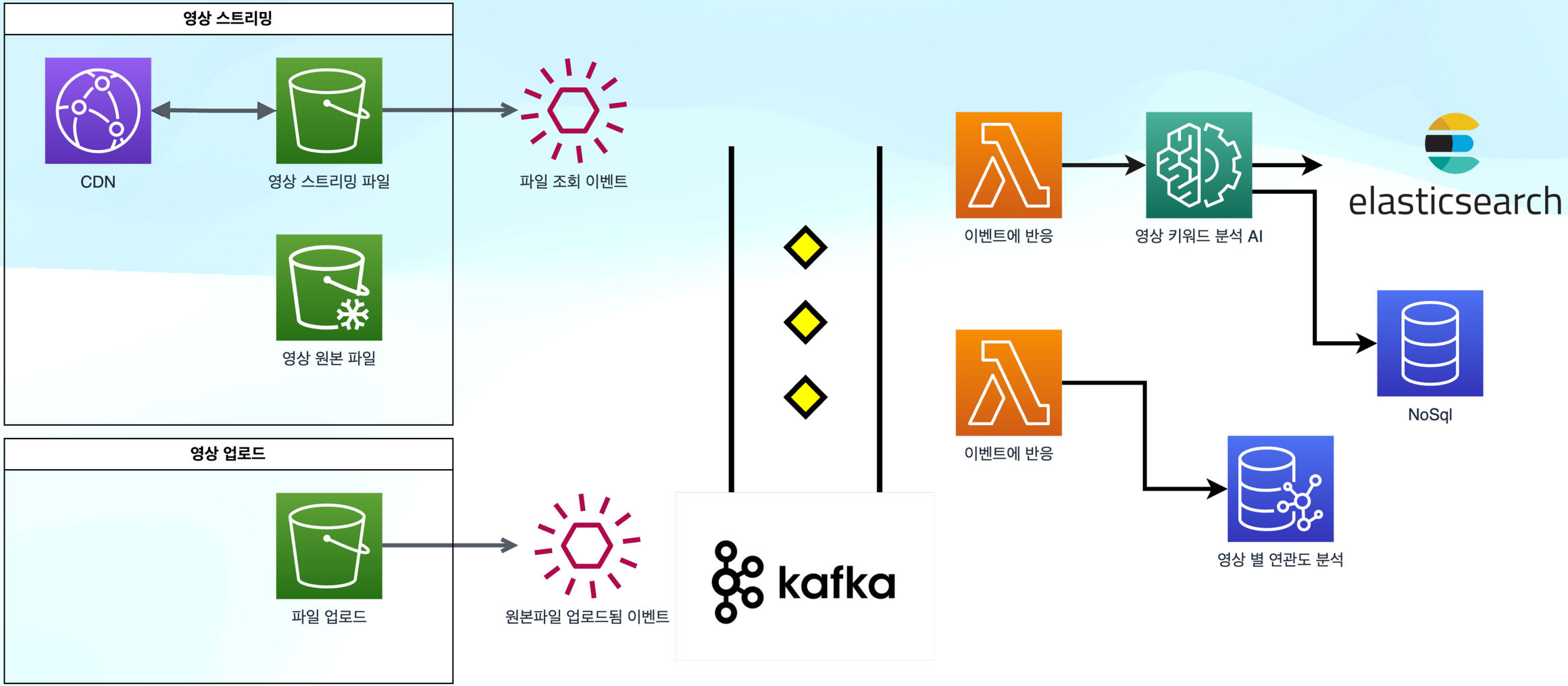





```
{
    "uploadURL": "https://s3uploader-s3uploadbucket-4557newpt3tc.s3.amazonaws.com/8634699.jpg?AWSAccessKeyId=ASIA3DTKMGNKFYZC2D6D&
        Content-Type=image%2Fjpeg&Expires=1594653821&Signature=hDsal1%2BOf5wXttmVX4iTG7ymmPc%3D&
        x-amz-security-token=IQoJb3JpZ2luX2VjEND%2F%2F%2F%2F%2F%2F%2F%2F%2F%2F%2FwEaCXVzLWVhc3QtMSJHMEUCIQC%2Bx8ItXEwYXiN%2Bu0RbzJfhj6Y
        5J3r4p5wNW6jornKENQIgOFDrZYez2RWOILkoesFitURFiLcLtXLDet01%2FV0msYcq7QEIeRABGgw3NjM2NTM1MzQ1NDgiDNzkftitsD642pecVSrKAdwvWFJ0%
        2FOryjiz86gjKu%2BeL0VJKGWEpmuVDGG3BUk4B0%2BWyx4BnaXzi3hxITwfcRXLFiYF6AmAEek5Sd9qrgEcwbXnb%2BLt2zWqSWdsooDTYLMxRAFfSn85NpKybe
        i5MZMo7SkAMkiJA1EJjvI9KdhfSI3Mm7y4maso8ceDa5iIXszx7ttIPrbnRcm8DmRSVZpwxLjAl5wQajT%2FOA%2BqB4ru%2BY4ZF%2BxRga6bGuc2DcLBkEmK3i
        eNaTdFiglbPYSgmifYMZAYmIEAsEpgwvvix%2BAU64AEN%2FHNwJ8%2BuLBtKqkVxxHVVWzVXW4p5iYXewlem91dfhyF7GogTvSlfLH8W1p3gIr0n9qG6OGNxGl%
        2BW91%2Bxzd6e6Ff94MNwvgJ823kXCP6mKASyrNNqikf82hyazzU8bjOlYpUJEsJqTyEroU1PJ%2BCUuuXhULeB%2FUoU0C41KJqcNxyIR7H3vhywwtZgnu80orXO
        A%2Fdad7IQ1th9dyiQqOS%2BLMl0qvihV1%2BS9ou0RUwkir1ADse90S8OLf7FgGrER0fgunUHxqzn79651WQT5FjRRIN%2FjQYRVFjO7kG2%2B9wzuuw%3D%3D"
        ,
    "Key": "8634699.jpg"
}
```

추후 개선 예정 사항

동영상 업로드 개선



“감사합니다”